Proceedings of the 2023 Winter Simulation Conference
C. G. Corlu, S. R. Hunter, H. Lam, B. S. Onggo, J. Shortle, and B. Biller, eds.

A NETWORK-BASED ANALYTICS FRAMEWORK FOR HIGH-RESOLUTION AGENT-BASED EPIDEMIC SIMULATION ENSEMBLES

Amro Alabsi Aljundi
Galen Harrison
Jiangzhuo Chen
Madhav V. Marathe
Henning Mortveit
Anil Vullikanti
Abhijin Adiga

Biocomplexity Institute
University of Virginia
Charlottesville, VA 22903, USA

ABSTRACT

High-resolution network-based contagion models are being increasingly used to study complex disease scenarios. Due to network-induced heterogeneity and sophisticated disease and intervention models, even simple simulation exercises can lead to large volumes of complex simulation outcomes. New approaches are required to analyze them. Simulations of such network spread processes can be viewed as attributed temporal graphs. We describe a network-based analytics framework that enables a user to leverage this graphical viewpoint and apply graph mining methods to perform fine-grained analysis of the simulation outcomes and the underlying network. The framework is based on a microservices-oriented architecture, and is designed to be general, adaptable, and scalable. We demonstrate its utility through a case study motivated by the COVID-19 pandemic involving the spread of two variants on a large realistic population network with multiple interventions. We study the transmissions within and between age-groups, importance of non-essential interactions, and efficacy of interventions.

1 INTRODUCTION

1.1 Background and Motivation

The evolution of an epidemic such as COVID-19 is a result of the infectious disease characteristics (virulence, variants, etc.) and the dynamics of the population affected by it (immunity levels, nature of interactions, epidemic response, etc.) (Qiu et al. 2022; Buckee et al. 2021; Verelst et al. 2016; Ferretti et al. 2020). Motivated by the availability of fine-grained data, there have been several works on using very high-resolution network-based dynamical systems to represent and study these complex disease dynamics scenarios (Abueg et al. 2020; Ferretti et al. 2020; Hoops et al. 2021; Kerr et al. 2021; Aleta et al. 2020; Chen et al. 2022). These bottom-up data-driven models use realistic representations (digital twins) of real-world populations captured by large node- and edge–attributed temporal socio-technical networks. They have been successfully applied to study various aspects of epidemic response such as vaccine allocation, contact tracing, and behavioral responses.

In the context of computational epidemiology (Marathe and Vullikanti 2013), such a network-based dynamical system comprises an underlying *contact network* that represents the interactions between individuals in a population, disease *spread model*, various *interventions*, and *scenarios* that specify the initiation and termination of events such as disease spread and policy-level responses during the course of the sim-

979-8-3503-6966-3/23/$31.00 ©2023 IEEE

ulation. The simulation output containing individual-level state transitions can be naturally and succinctly represented as a temporal graph capturing states of individuals at relevant time steps and who-infected-who information. We refer to this as a *cascade network* or simply a cascade (Newman 2003).

Large-scale graph analytics is an active area of research. There are several works in this area addressing system design, visualization, and algorithmic challenges (Batarfi et al. 2015; Sahu et al. 2020; Liu et al. 2020). This paper focuses on applying graph analytics in the context of network-based contagion modeling of large populations and the graphical outputs they produce. Simulations play an important role in the analysis of such models. Even simple dynamical properties such as expected outbreak size are computationally hard to determine; the only viable approach is to perform large-scale simulation analysis. See, for example, works on evaluation of contact-tracing protocols in Aleta et al. (2020) or Hoops et al. (2021), interventions allocation (Chen et al. 2022; Kerr et al. 2021). Simulation analytics in this setting is challenging from several perspectives. Firstly, owing to the size of the contact networks, even simple disease dynamics scenarios lead to the generation of large attributed cascade graphs. Besides network induced heterogeneity, realistic scenarios of disease spread include complex multi-resolution multi-type interventions and behavioral responses. Together, this combination leads to enormous amounts of simulation data (graph data like cascades and contact networks) presenting computational and algorithmic challenges for generating simulation statistics, synthesis, and visualization.

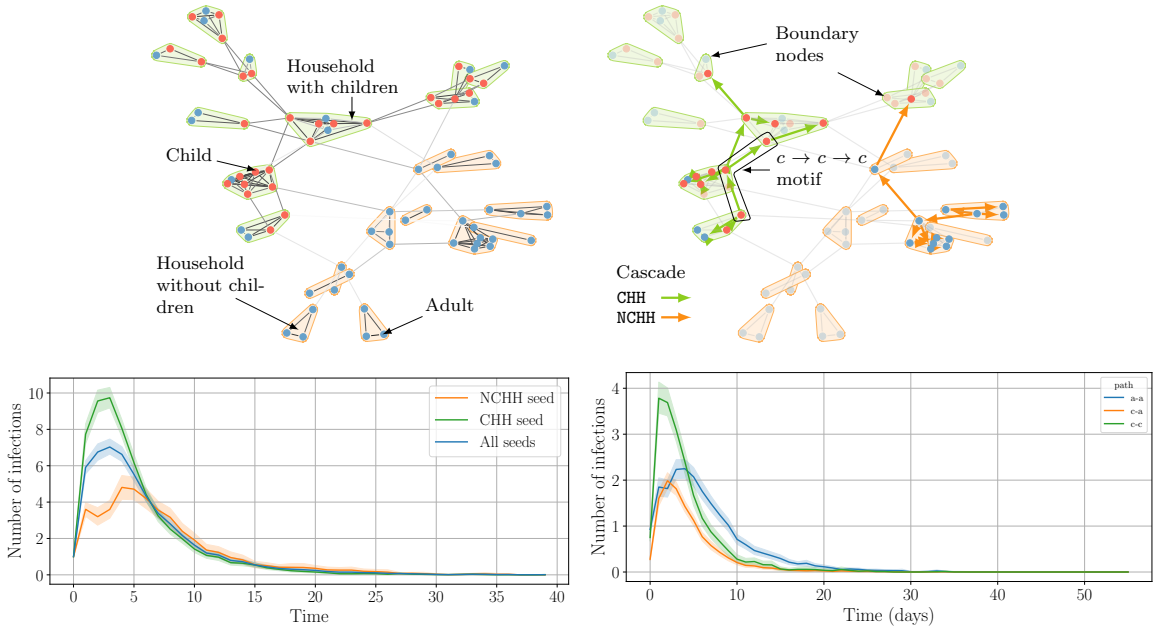

Figure 1: A motivating example. (a) A contact network (top-left) and (b) two cascades (top-right) are shown for a small realistic subpopulation. There are two types of households, those with children (CHH) and those without (NCHH). Nodes are labeled by age-group (adult or child). The edges are weighted by the duration of interaction (darker means longer). The diffusion process is a simple SIR model with a transmission rate 5×10^{-5} and a recovery rate 0.2 with a single seed node chosen randomly. The bottom-left plot shows that the dynamics strongly depend on which type of household the seed node belongs to, which is not captured by the average number of infections per time step. The bottom-right plot shows that child nodes and more generally, CHH households strongly influence the spread of the disease. These are the top three transmission motifs considering that children account for only 30% of the population. The Pearson's correlation coefficient for the epicurve when compared with the labeled edge count time-series across cascades are as follows: $c \to a$: 0.83, $c \to c$: 0.77, while $a \to a$ is only 0.65.

1.2 Contributions

This paper proposes a design process for graph analytics of high-resolution network-based simulations in the context of epidemics. In this work, we view simulation outputs from high-resolution network dynamical systems as cascade graph ensembles, where each cascade is a highly structured graph with rich node and edge attributes that are inherited from the network and the dynamics. This graphical view of the simulations enables us to perform a fine-grained analysis that can provide insights into the complex interplay between individual-level interactions and disease propagation, which would not be possible with simpler measures. Harrison et al. (2023) demonstrate the importance of using such structural features of cascades in characterizing complex disease scenarios. Such fine-grained analysis requires mining rich structural and relational patterns from the cascade graphs and the contact networks. Figure 1 illustrates this point. One can take advantage of graph mining methods to discover higher order patterns. Thirdly, this viewpoint is timely given the emergence of mobile tracing technologies that have the ability to collect fine-grained data on the evolution of the disease or the cascade (Ahmed et al. 2020; Anglemyer et al. 2020; Ferretti et al. 2020; Colizza et al. 2021; Rodríguez et al. 2021; Tsvyatkova et al. 2022).

Viewing network diffusion processes as graphs is not new. It is routinely applied to various computational problems such as inference of cascades and control of diffusion processes (Rozenshtein et al. 2016; Shah and Zaman 2011). However, these works are limited to very simple scenarios (e.g., independent cascades on a simple undirected network). To the best of our knowledge, this is a rarely used point of view in the analysis of high-resolution ABM models. The statistics collected from the simulations is limited to counts of individuals or groups (infected, belonging to a certain age group, vaccinated, etc.) and their interactions.

We propose a graph simulation analytics design that incorporates microservices-oriented and pipes & filters architectural styles. This approach is motivated by two requirements. First, there is a need to integrate a wide variety of network measures considering their implementations and required computing infrastructure. Second, to cater to a variety of studies and simulation environments, there is a need for generality and adaptability. Other important considerations are transparency, replicability, auditability, ease of knowledge transfer, scalability and reduction of errors. Our system is loosely tied to the study and the simulation environment. It only expects the simulation outputs to be of a certain form so as to be able to construct cascade graphs. This enables the user to specify the study- and simulator-specific graph features and compute a chosen set of cascade graph properties. Accordingly, the framework is composed of four subsystems (Figure 2): (i) Cascade graph construction and augmentation; (ii) Cascade graph measures and mining; (iii) Contact network measures and mining; and (iv) Consolidation of measures. Each subsystem is a pipeline framework consisting of several pipelines, which in most cases can be executed in parallel.

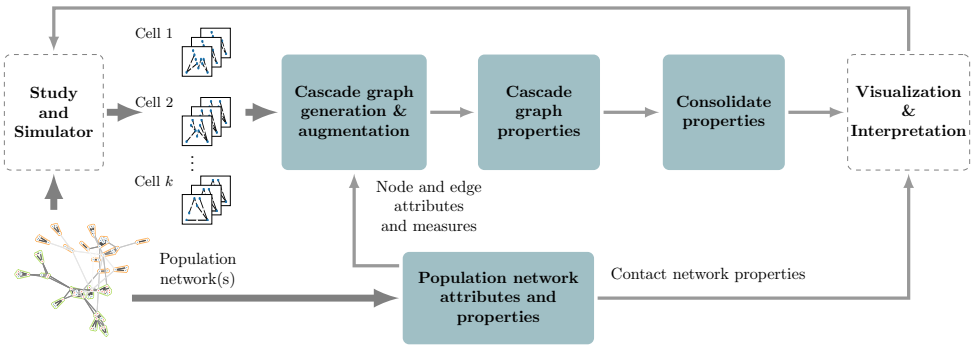

Figure 2: The pipeline framework for simulation analytics.

We showcase the utility of the graph analytics framework through a case study motivated by the COVID-19 pandemic. The contact network is derived from a large realistic synthetic population dataset of a US county. The spread model is a popular model of COVID-19. The scenario involves two variants, several

interventions such as vaccination, social distancing, and school/work closure. We study the transmissions within and between age-groups during the course of the epidemic. Our analysis of interactions by activity types shows the importance of non-essential activity types (that induce weak links) in the spread of the disease. We note that in the presence of a highly infectious variant, limiting only non-essential activities might not be effective. We also investigate the correlation between superspreader events (a node infecting several nodes) and spikes in the epicurve.

2 DISEASE CASCADES AND THEIR STRUCTURAL MEASURES

2.1 Contact Graph and Diffusion Model

Let $G_P(V_P, E_P)$ be a contact graph corresponding to the study population with node set V_P and edge set E_P on which the disease diffusion occurs. It is undirected and has multi-edges, where each edge has a label and weight associated with it. For example, a label type could be induced by the activities of the endpoints that led to the interaction; a label `work-work` means that both nodes interacted during work activity, while `home-work` means that both nodes interacted when one node was at its home while the other was at work in the former's residence location. Each edge weight (w_e for an edge e) corresponds to the time duration of the interaction. An example network is illustrated in Figure 1(a). The network was extracted from a large contact graph representing the state of Virginia (Harrison et al. 2023) as an induced graph of nodes in an urban area. We observe that the households in `CHH` have a higher interaction within this category than outside primarily due to interactions between children.

2.2 Diffusion Model and Simulation Instances

Our current work is applicable to a broad class of discrete-time diffusion models on networks (Marathe and Vullikanti 2013; Easley and Kleinberg 2010; Prakash et al. 2012). Broadly, a node can be in four classes of states: **S**, **E**, **I**, or **R**. A susceptible node v (node state **S**) is infected by an infectious neighbor u (node state **I**) probabilistically, and transitions to the exposed (**E**) state. Nodes in **E** state transition to the infectious (**I**) state after a dwell time. Nodes in **I** state transition to the recovered (**R**) state again after a dwell time. Depending on the model, nodes can get reinfected (say by a different variant). This is modeled by introducing variant-specific states. A simulation output can be succinctly represented as a set of tuples denoting state transition events of the form $\big(v, t, \text{state}(v,t), v'\big)$ where t is the time step, v is the node which transitioned to a new state $\text{state}(v,t)$ at time t, and v' is a neighbor that infected v at time t. For all transitions except $\mathbf{S} \to \mathbf{E}$, $v' = \perp$ denoting that there is no parent. A *seed node* is a node that has no parent. The disease spread starts from a set of seed nodes but the seeding can occur at later time steps as well. The event of choosing a node v as the seed at time t is denoted by $(v, t, I, \perp)$. In general, our methods are relevant beyond SEI(R)S models. We are using this model class as an example.

2.3 Cascade Graph

Supposing that a disease diffusion process is observed for time steps $0, \ldots, T$ on the contact graph $G_P(V_P, E_P)$, where T denotes the *time horizon*. The evolution of the diffusion process can be captured through a *disease cascade graph* $C(\mathscr{V}, \mathscr{E})$ whose node set $\mathscr{V} \subseteq V \times [T]$ is the *time-expanded* version of V_P, where each node $(v,t) \in \mathscr{V}$ denotes that v was exposed at time t. Its edge set $\mathscr{E}$ corresponds to the set of directed edges $\big((v',t'),(v,t)\big)$, where $t' < t$ denoting that v' infected v at time t. Every $\mathbf{S} \to \mathbf{E}$ state transition $\big(v, t, \text{state}(v,t), v'\big)$ event is mapped to an edge in E. Note that each cascade graph is a directed acyclic graph (DAG). In the case of SEIR model, a node is infected at most once. Hence, in some instances, we use a simpler version of the node and edge set by stripping the time information. Let $V = \{v \mid (v,t) \in \mathscr{V}\}$ denote the set of nodes in V_P that were infected in C and $E = \{(v', v) \mid \big((v',t'),(v,t)\big) \in \mathscr{E}\}$ denote the set of edges on which transmissions occurred. Let $\mathscr{C}$ denote the set of all valid cascade graphs. Two example cascades are shown in Figure 1(b), each seeded by one node for the same diffusion model. The first

cascade C_1 is seeded by a node in CHH while C_2 is seeded by a node in NCHH. Transmission is modeled by the Direct Gillespie Method, and hence, the resulting cascade is a tree (every infected non-seed node has exactly one parent).

2.4 Structural Measures and Their Significance

Structural properties are computed for both the cascades as well as the population networks. Some measures are listed in Table 1. From the perspective of granularity, the measures can be aggregated at different levels: node- or edge-level, group-level, and graph-level. Node- and edge-level measures capture the importance of specific nodes across cascades. These include structural measures such as degree, centrality, and clustering coefficient as well as cross-cascade properties such as number of cascades in which a node was infected, which is an empirical estimate of its vulnerability. A group-level measure is computed with respect to a group or set of nodes (or edges). An example is a labeled path motif, where labels can be either age groups (e.g., child, adult) or occupations (e.g., health care, caregivers). Group-level properties are obtained by aggregating over all nodes or edges grouped by specified features. Graph-level features are single-valued parameters such as diameter or distributions of degree. Network embedding algorithms can be used to generate graph-level and subgraph-level embeddings. Note that epidemic measures such as the total number of infections per time can be naturally expressed as cascade properties.

Table 1: Some structural properties that can be computed on the cascade graph and the contact network.

Name	**Description**
Cascade graph properties	
Num. of nodes	Number of nodes w.r.t some feature(s), e.g., number of infected nodes per (i) time step (epicurve), (ii) age group, or (iii) by state.
Num. of edges	Number of live edges w.r.t some feature(s), e.g., number of live edges by activity.
Level	Distance from seed node(s) (w.r.t. time)
Out degree	Number of neighbors infected by a node.
In degree	Number of neighbors infecting a node.
Node- and edge-labeled motifs	Counts of paths and tree motifs. For e.g., a directed path $c \xrightarrow{s} c \xrightarrow{h} a$ corresponding to a child infecting another child in school, who in turn infects an adult in its household.
Num. of boundary nodes	Number of uninfected nodes that are neighbors of nodes in the cascade.
Boundary node degree	Number of infected neighbors in the cascade.
Vulnerability of a node	Number of cascades in which a node is infected.
Average time of infection	The average time of infection of a node. For example, this is useful for developing strategies for early detection.
Subgraph embeddings	Low dimensional embeddings of cascades or their connected components.
Contact network properties	
Num. of nodes and edges	Number of nodes or edges w.r.t. some features.
k-core and core number	The k-core is the maximal induced subgraph with minimum degree at least k.
Spectral properties	Eigenvalues and eigenvectors of adjacency and Laplacian matrices.
Centrality measures	Betweenness centrality, eigen centrality, degree centrality, etc.
Degree	Degree and weighted degree distributions, top degree nodes, k-hop degree,etc.
Node- and edge-labeled motifs	Similar to cascade graphs.
Clustering coefficient	Local and average clustering coefficients.
Distances	Diameter, eccentricity, radius, etc.

3 NETWORK-BASED SIMULATION ANALYTICS FRAMEWORK

The design process for our graph analytics framework follows recent works in the context of large-scale data driven simulation systems (Thorve et al. 2022). We use two design paradigms: microservices-oriented architecture (MSA) (Richardson 2018; Wolff 2016) and pipes and filters architecture (Bass et al. 2003; Buschmann et al. 2008). The design process anticipates the need to integrate a diverse set of network measures, address different types of questions that arise in modeling exercises as responders seek to answer

complex questions efficiently, and effectively leverage the varied expertise of cross-functional teams in a collaborative environment. MSA consists of loosely coupled, autonomous, and specialized microservices or *h*-functions. The pipes and filters pattern decomposes tasks into a series of functions or filters which are composed in a pipeline. In our case, filters are microservices which encapsulate a functionality and pipes serve as connectors to pass data streams from one filter to another. Each such *h*-function corresponds to a processing step scoped out in such a manner that it can be reused in different contexts, and can be extended independent of the system. This architecture is suited for the simulation analytics framework in multiple aspects.

Firstly, a plethora of network science and graph mining tools are available which vary widely in software implementation, computing environment, and contributing developers. Examples are provided below in the description of system. This heterogeneity can be handled by the modular approach of the architecture. Secondly, it also avoids a "sea of components" (Buschmann et al. 2008) as functions can be reused multiple times within and across pipes as required. Specialized teams can independently update or extend a function with minimal disruptions to the system as a whole. Thirdly, it facilitates the parallelization of tasks and efficient memory management.

The spread model used and the measures to be computed depend on the questions being addressed, which vary greatly across studies. The augmented cascade network described in Section 2.3 is general enough to handle these demands. Our case study in Section 4 demonstrates the complex scenarios that this application can cater to.

The pipelines in the subsystems can be reconfigured to a considerable extent by the user via configuration files. This can reduce the time taken to ramp up to a study and also effectively reuse code. The architecture is also suited for exploratory studies, debugging, and verification & validation exercises. These are typically iterative processes where the entire system needs to be rerun multiple times. The application allows for running specific measures and retaining intermediate derived data to hasten the process at each stage.

Our simulation analytics system is illustrated in Figure 2, and consists of four main subsystems: (i) Population network attributes and measures (POPNET), (ii) Cascade graph generation and augmentation (CASGEN), (iii) Cascade graph properties (CASPROPS), and (iv) Consolidation of measures (CONSOLIDATE). Each subsystem is a pipeline framework that handles tasks that can be accomplished as a composition of microservices. We will describe each subsystem in detail. There are two other systems that are illustrated in Figure 2: Study & Simulator (S&S) and Downstream visualization and interpretation (VIZINT). These two systems are integral part of any simulation analysis exercise. However, they are deliberately decoupled from the simulation analytics application for the following reasons. Firstly, our application is general enough to support multiple studies and simulation systems as long as the simulation outputs are of a certain form (like the one defined in Section 2.1) and the questions can be addressed by computing graph properties. Secondly, once the properties of the cascades and the population network are made available, various statistical and machine learning tools can be applied to interpret the results.

The system S&Sgenerates the three main inputs that drive our application, namely $c_{\text{SS}}, \mathscr{S}$, and $\mathscr{G}$. The input c_{SS} is an experiment configuration specification (JSON schema) that describes the simulation model and experiment scenarios or cells that generated the various simulation replicates. Each simulation output $S \in \mathscr{S}$ is as described in Section 2.1 with metadata inherited from the cell information described in c_{SS}. Next, we have a collection of population networks $\mathscr{G} = \{G_P^1, G_P^2, \ldots\}$ on which the diffusion processes were run. Each population network is specified by two mandatory datasets, one for nodes and their features and the other for edges and their features. More features can be specified through tables.

3.1 Cascade Graph Generation and Augmentation

This subsystem denoted by $\text{CASGEN}(c_{\text{CG}}, \mathscr{S}, \mathscr{F}_{\text{PN}}, c_{\text{CP}})$, constructs cascade networks from the simulation outputs $\mathscr{S}$ and population network features $\mathscr{F}_{\text{PN}}$ (see Figure 3), and uses c_{CP}, the configuration input file for the CASPROPS framework to store the results in the required formats. The configuration file c_{CG} (JSON) specifies the meta data, simulation instance, and population network properties to generate each cascade

network. For each $S \in \mathscr{S}$, this subsystem computes the corresponding cascade network $C \in \mathscr{C}$, the set of cascade networks. There are functions to convert and store this data into formats required for downstream tasks (like CSV or Feather for Pandas and Networkx (Hagberg et al. 2008) or TTables for SNAP (Leskovec and Sosič 2016)). These functions are invoked based on what measures need to be computed, which is specified in c_{CP}.

3.2 Population Network Attributes and Measures

This subsystem has two objectives: (i) it provides population network data to the CASGEN module to generate the cascade graphs and (ii) it computes various network measures in the form required by CASGEN. It is denoted by POPNET$(c_{\text{PN}}, \mathscr{G})$, where c_{PN} is a configuration file (JSON schema) that provides task specifications. Since it has similarities to the CASPROPS module (and due to space constraints), we have not illustrated this module. There are several types of microservices in this framework. The first set of microservices corresponds to the manipulation of the various input data frames such as selecting required features and joining data frames that provide input to CASGEN and other functions within this framework. The second set of microservices corresponds to computing network measures of various types as described in Section 2.4. A third set of microservices converts data from one format to another as in the previous module. The outputs of this function are a set $\mathscr{F}_{\text{PN}}$ of node- and edge-level features in data frames for each contact network in $\mathscr{G}$ and a set $\mathscr{P}_{\text{PN}}$ of network measures.

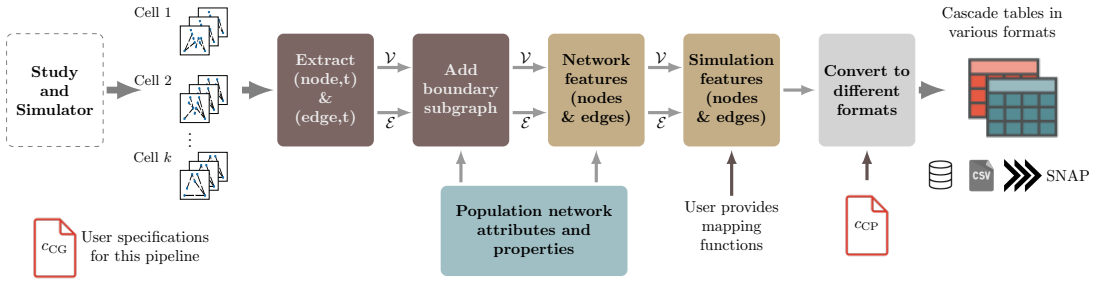

Figure 3: The cascade graph generation and augmentation subsystem CASGEN.

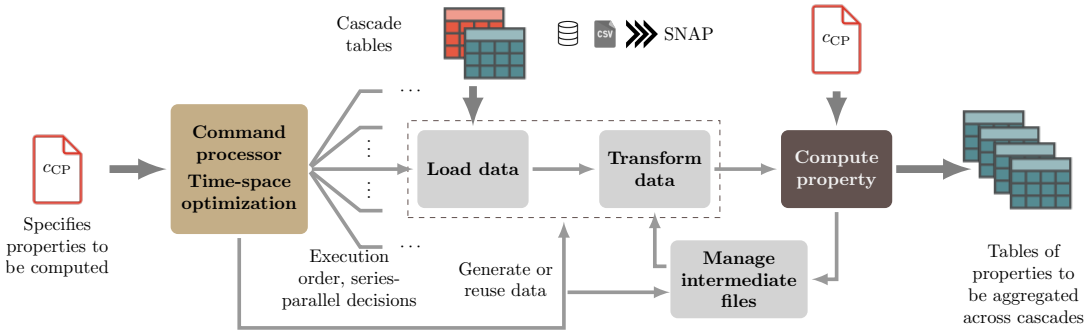

Figure 4: Cascade graph properties subsystem CASPROPS.

3.3 Cascade Graph Properties

The subsystem CASPROPS$(c_{\text{CP}}, \mathscr{C})$ computes various network measures for the cascade graphs. Architecturally, it is similar to the POPNET subsystem. Most microservices corresponding to the manipulation of data frames and data format conversion can be reused in this subsystem. There are several measures common to both subsystems. In principle, functions from POPNET corresponding to these measures can

be used here as well. However, the fact that cascade networks are DAGs can be leveraged to efficiently compute many of these measures, e.g. labeled path motif counts can be computed in linear time for DAGs, while these counts are computationally hard to compute for arbitrary graphs. Hence, many microservices corresponding to measures are different from those in the POPNET subsystem. Meta data is generated for each cascade containing experiment cell data. For each cascade network $C_i \in \mathscr{C}$, let $\mathscr{M}_i$ denote the set of corresponding output data frames and let $\mathscr{M} = \bigcup_i \mathscr{M}_i$.

3.4 Consolidation of Measures

This module denoted by CONSOLIDATE$(c_{\text{CP}}, \mathscr{M})$ collects and combines data from all cascades for efficient storage and convenient access to downstream tasks. Node- and edge-level properties are aggregated across cascades corresponding to the same cell. Group-level and graph-level properties are concatenated together along with cascade and cell information. The configuration file c_{CP} from CASPROPS informs how aggregation happens across cascades. This module is not illustrated for space reasons.

4 CASE STUDY

This case study, motivated by the COVID-19 pandemic demonstrates the value of the proposed simulation analytics application. We consider a complex but realistic scenario of an infectious disease spreading in a large population based on a popular COVID-19 disease model (Centers for Disease Control and Prevention 2020; Chen et al. 2022). The scenario described below includes multiple variants, age-based infectivity and susceptibility, waning immunity, and various pharmaceutical and non-pharmaceutical interventions such as school/work closure, vaccination, and generic social distancing triggered at different points of time. Through the lens of various structural measures of the cascades at different phases of the simulation we address several topics: (i) the effect of school reopening on disease spread, (ii) the spread in a household, (iii) the effect of social distancing, (iv) two variants, and (v) effect of vaccination.

For this case study, we used a synthetic contact network meant to represent social contacts within Montgomery County, Virginia. This network was developed by combining census data with measurements of mobility sourced from public and commercial data sets (PREPARE 2023). This network is composed of around 83312 individuals (nodes) and 3,326,934 million edges. The details of the construction of this network can be found in Harrison et al. (2023). The network represents a normative weekday in a year. Nodes may have multiple edges between one another, indicating multiple contacts at different times of day.

The scenario consists of a sequence of events, each implemented as a major structural change in the population network or a change in the characteristics of the diffusion model.: (i) Jan. 31, 2020: simulation/pandemic started; (ii) Mar. 16, 2020: population lockdown / stay-home mandate is ordered; (iii) Jun. 14, 2020: lockdown is lifted, schools remain closed, 40% of population complied with generic social distancing; (iv) Nov. 26, 2020: new variant started; (v) Jan. 15, 2021: vaccination started in 65+; (vi) Feb. 24, 2021: generic social distancing is relaxed to 20% compliance; (vii) Mar. 06, 2021: vaccination started in 18–64 too; (viii) Mar. 16, 2021: schools reopened; and (ix) May 25, 2021: simulation ends. Pharmaceutical interventions (PIs) such as vaccination change node susceptibility and/or infectivity; and non-pharmaceutical interventions (NPIs), such as social distancing, change node behavior and as a result change edges of the contact graph. Vaccination reduces the susceptibility of a compliant node by a fraction, which is called the vaccine efficacy (VE). This reduces the probability that the node will be infected by other nodes. Generic social distancing (GSD) removes all non-essential activities of a compliant node. The essential activities include `home, work, school` while the nonessential activities include `shop, other`. Since every edge is associated with two activities, one for each end point, there are three categories of edges: essential–essential (`E`), non-essential–non-essential (`N`) and essential–non-essential (`M`). All type-`N` edges incident with the compliant nodes and those type-`M` edges where the non-essential side belongs to a compliant node will be removed from the contact graph. The results are presented for 100 replicates.

4.1 Results

The population is divided into five age groups: preschool (p: 0-4), students (s: 5-17), adults (a: 18-49), old (o: 50-64), and seniors (g: 65+). In Figure 5, the top three labeled edge counts are provided, which involve only two age-groups: a and s. We observe that during the lockdown, most of the infection transmissions are due $a \to a$ transmissions. The transmissions $a \to s$ are mainly within-household occurrences. The $s \to s$ transmissions are primarily dependent on school activities. We see that soon after schools reopen, a large uninfected and unvaccinated population of students interact and lead to outbreaks. Note that the $a \to a$ transmission that was decreasing plateaus rapidly during the surge of $s \to s$ transmissions (time steps 400–430). This is due to $s \to a \to a$ paths of infections (not shown here) within households.

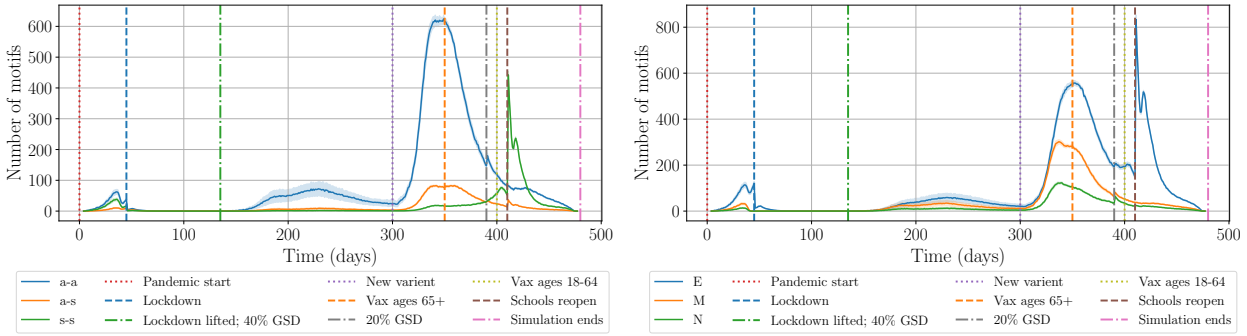

Figure 5: Transmission of infections due to different types of interactions are shown for different phases of the epidemic. The plot on the right corresponds to top three age-group based path motifs of length 1 (or labeled edges). The plot on the left corresponds to dominant activity based path motifs of length 1.

Even though a large portion of the transmissions is due to essential interactions, in Figure 5 (right), we observe the efficacy of GSD (and in turn the importance of non-essential edges) when the GSD level is reduced by half at around time step 390. We note that the number of $a \to a$ infections, which is decreasing rapidly plateaus. This is due to the extra connectivity provided by non-essential activities.

By design, the second variant that we considered is more infectious than the first one. Due to the lockdown, high GSD, and lower infectivity, the spread due to the first variant is reduced to a few isolated events even after the lockdown is lifted (200–300). In the absence of lockdown, the second variant leads to an outbreak despite high GSD, indicating that beyond a certain transmissibility, GSD is ineffective as non-essential edges are typically fewer and of shorter duration (lower edge weights). This is partially explained by the threshold phenomenon that is observed in SIR(S)-like processes: for many networks, there exists a certain transmissibility threshold, below which the outbreak sizes are negligible, and above which they are large (Prakash et al. 2012).

Our work demonstrates the use of simulations for evaluating decisions such as relaxing social distancing levels and school reopening. Even though the infection size is going down rapidly just before school reopening, we observed an outbreak soon after. At the same time, we observe that lifting the lockdown does not lead to a substantial number of infections.

We characterize a superspreader event (SE) as a node infecting three or more people, which corresponds to a node in the cascade having an out-degree at least 3. In Figure 6 left plot, we observe a sharp peak in the SE events (time 300 to 350) for adults right before the peak in the epicurve (around 370), and a sharp peak in SE events for children (400–425) before the second peak in epicurve. These gaps are quantified in the center and right plots respectively. Comparing the sharpness of the peaks of the SE events to those of the epicurves show that relatively very few simultaneous SE events can lead to a large number of infections per time step sustained over a long period, due to cascading effects. This analysis suggests that preventing SE events not only helps reduce or prevent the peaking of infections in the epicurve but also significantly reduce the total number of infections.

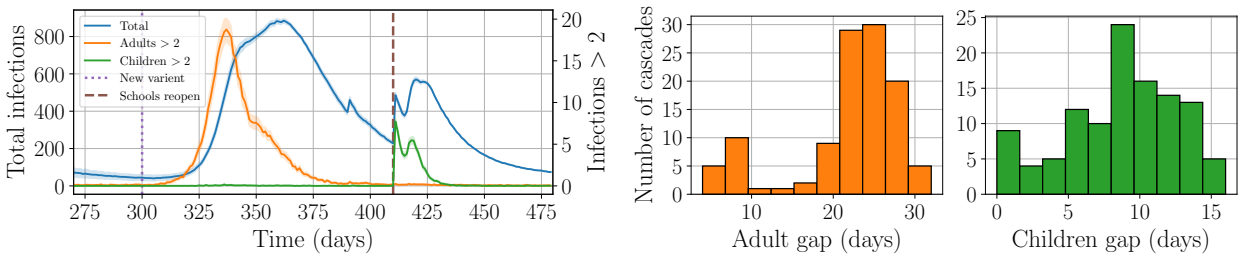

Figure 6: Temporal relationship between superspreader events (SE) and community-wide infection peaks. The center figure measures the gap between the peaks of epicurve and adult SEs in the period $[300, 400]$, and the right figure measures the gap between the peaks of epicurve and children SEs in the period $[400, 480]$.

5 RELATED WORK

Prior work, particularly in the context of epidemiological modeling, has emphasized the role of network structure and its effects on aggregate outcomes. For example, Aleta et al. (2020), Chen et al. (2022), all investigate the effects of manipulating different aspects of the network on the overall transmission dynamics (e.g. total infected or the number of new infections over time). In particular, Chen et al. (2022) find that node degree works well as a vaccination strategy. These lines of work are interested in questions of the form "given a change in the network structure (possibly expressed as model parameters), what are the aggregate effects?" The question we are primarily interested in is to what extent changes in model parameters lead to differences in the structure of how infection spreads. This question has received somewhat less attention. While some simulation frameworks (e.g. CovaSim (Kerr et al. 2021)) enable these types of queries, they largely leave open the question of what the relevant queries are and how to efficiently execute them.

Similarly, while large-scale graph analytics frameworks have received quite a bit of attention (Batarfi et al. 2015; Sahu et al. 2020; Liu et al. 2020), the focus has been on structural aspects of arbitrary graphs. For instance, methods for efficiently querying properties such as connected components and motifs have been developed (Batarfi et al. 2015; Liu et al. 2020). Simulation cascades on the other hand are highly structured graphs. None of the large-scale graph analytics platform provide support for the analysis of such graph ensembles in the context of network dynamical systems.

Even though large-scale adoption of digital contact tracing occurred during COVID-19 (Ahmed et al. 2020; Anglemyer et al. 2020; Alo et al. 2022; Colizza et al. 2021; Wymant et al. 2021; Moosa et al. 2023), there is not much publicly available structural data for analysis. However, there have been some recent model-based studies (Kretzschmar et al. 2020; Aleta et al. 2020; Kucharski et al. 2020) that have modeled and evaluated contact tracing. While our work does not evaluate tracing, our experimental results show the importance of such data for analysis. In a much simpler setting, Harrison et al. (2023) quantify the importance of structural features in learning complex disease scenarios.

ACKNOWLEDGMENTS

This work was partially supported by National Science Foundation (NSF) RAPID Grant No. CCF-2142997, University of Virginia Strategic Investment Fund award number SIF160, NIH Grant 1R01GM109718, NSF Grant OAC-1916805 (CINES), NSF Expeditions Grant CCF-1918656, and FACT grant 2019-67021-29933 from the USDA National Institute of Food and Agriculture.

REFERENCES

Abueg, M., R. Hinch, N. Wu, L. Liu, W. Probert, A. Wu, P. Eastham, Y. Shafi, M. Rosencrantz, M. Dikovsky et al. 2020. "Modeling the Combined Effect of Digital Exposure Notification and Non-Pharmaceutical Interventions on the COVID-19 Epidemic in Washington State". *MedRxiv*:2020–08.

Ahmed, N., R. A. Michelin, W. Xue, S. Ruj, R. Malaney, S. S. Kanhere, A. Seneviratne, W. Hu, H. Janicke, and S. K. Jha. 2020. "A Survey of COVID-19 Contact Tracing Apps". *IEEE Access* 8:134577–134601.

Aleta, A., D. Martin-Corral, A. Pastore y Piontti, M. Ajelli, M. Litvinova, M. Chinazzi, N. E. Dean, M. E. Halloran, I. M. Longini Jr, S. Merler et al. 2020, September. "Modelling the Impact of Testing, Contact Tracing and Household Quarantine on Second Waves of COVID-19". *Nature Human Behaviour* 4(9):964–971.

Alo, U. R., F. O. Nkwo, H. F. Nweke, I. I. Achi, and H. A. Okemiri. 2022. "Non-Pharmaceutical Interventions Against COVID-19 Pandemic: Review of Contact Tracing and Social Distancing Technologies, Protocols, Apps, Security and Open Research Directions". *Sensors* 22(1):280.

Anglemyer, A., T. H. Moore, L. Parker, T. Chambers, A. Grady, K. Chiu, M. Parry, M. Wilczynska, E. Flemyng, and L. Bero. 2020. "Digital Contact Tracing Technologies in Epidemics: A Rapid Review". *Cochrane Database of Systematic Reviews* 8.

Bass, L., P. Clements, and R. Kazman. 2003. *Software Architecture in Practice*. Addison-Wesley Professional.

Batarfi, O., R. E. Shawi, A. G. Fayoumi, R. Nouri, S.-M.-R. Beheshti, A. Barnawi, and S. Sakr. 2015. "Large Scale Graph Processing Systems: Survey and an Experimental Evaluation". *Cluster Computing* 18:1189–1213.

Buckee, C., A. Noor, and L. Sattenspiel. 2021. "Thinking Clearly About Social Aspects of Infectious Disease Transmission". *Nature* 595(7866):205–213.

Buschmann, F., R. Meunier, H. Rohnert, P. Sommerlad, and M. Stal. 2008. *Pattern-Oriented Software Architecture: A System of Patterns*, Volume 1. John wiley & sons.

Chen, J., S. Hoops, A. Marathe, H. Mortveit, B. Lewis, S. Venkatramanan, A. Haddadan, P. Bhattacharya, A. Adiga, A. Vullikanti et al. 2022. "Effective Social Network-Based Allocation of COVID-19 Vaccines". *Proceedings of the KDD Health Day*.

Colizza, V., E. Grill, R. Mikolajczyk, C. Cattuto, A. Kucharski, S. Riley, M. Kendall, K. Lythgoe, D. Bonsall, C. Wymant et al. 2021. "Time to Evaluate COVID-19 Contact-Tracing Apps". *Nature Medicine* 27:361–362.

Easley, D., and J. Kleinberg. 2010. *Networks, Crowds and Markets: Reasoning About a Highly Connected World*. New York, NY: Cambridge University Press.

Ferretti, L., C. Wymant, M. Kendall, L. Zhao, A. Nurtay, L. Abeler-Dörner, M. Parker, D. Bonsall, and C. Fraser. 2020. "Quantifying SARS-CoV-2 Transmission Suggests Epidemic Control with Digital Contact Tracing". *Science* 368(6491):eabb6936.

Centers for Disease Control and Prevention 2020. "COVID-19 Pandemic Planning Scenarios.". https://www.cdc.gov/coronavirus/2019-ncov/hcp/planning-scenarios-h.pdf, accessed 30th March 2023.

Hagberg, A., P. Swart, and D. S Chult. 2008. "Exploring Network Structure, Dynamics, and Function Using NetworkX". Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States).

Harrison, G., A. Alabsi Aljundi, J. Chen, S. S. Ravi, A. Vullikanti, M. Marathe, and A. Adiga. 2023. "Identifying Complicated Contagion Scenarios from Cascade Data". In *Proceedings of the 29th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Harrison, G., J. Chen, H. Mortveit, S. Hoops, P. Porebski, D. Xie, M. Wilson, P. Bhattacharya, A. Vullikanti, L. Xiong, and M. Marathe. 2023. "Synthetic Data to Support US-UK Prize Challenge for Developing Privacy Enhancing Methods: Predicting Individual Infection Risk During a Pandemic". [Data set].

Hoops, S., J. Chen, A. Adiga, B. Lewis, H. Mortveit, H. Baek, M. Wilson, D. Xie, S. Swarup, S. Venkatramanan et al. 2021. "High Performance Agent-Based Modeling to Study Realistic Contact Tracing Protocols". In *2021 Winter Simulation Conference (WSC)*, edited by K. Sojung, B. Feng, K. Smith, S. Masoud, and Z. Zheng, 1–12. IEEE.

Kerr, C. C., R. M. Stuart, D. Mistry, R. G. Abeysuriya, K. Rosenfeld, G. R. Hart, R. C. Núñez, J. A. Cohen, P. Selvaraj, B. Hagedorn et al. 2021. "Covasim: An Agent-Based Model of COVID-19 Dynamics and Interventions". *PLOS Computational Biology* 17(7):e1009149.

Kretzschmar, M. E., G. Rozhnova, M. C. Bootsma, M. van Boven, J. H. van de Wijgert, and M. J. Bonten. 2020. "Impact of Delays on Effectiveness of Contact Tracing Strategies for COVID-19: A Modelling Study". *The Lancet Public Health* 5(8):e452–e459.

Kucharski, A. J., P. Klepac, A. J. K. Conlan, S. M. Kissler, M. L. Tang, H. Fry, J. R. Gog, W. J. Edmunds, J. C. Emery, G. Medley, J. D. Munday et al. 2020. "Effectiveness of Isolation, Testing, Contact Tracing, and Physical Distancing on Reducing Transmission of SARS-CoV-2 in Different Settings: A Mathematical Modelling Study". *The Lancet Infectious Diseases* 20(10):1151–1160.

Leskovec, J., and R. Sosič. 2016. "SNAP: A General-Purpose Network Analysis and Graph-Mining Library". *ACM Transactions on Intelligent Systems and Technology (TIST)* 8(1):1.

Liu, N., D.-s. Li, Y.-m. Zhang, and X.-l. Li. 2020. "Large-Scale Graph Processing Systems: A Survey". *Frontiers of Information Technology & Electronic Engineering* 21(3):384–404.

Marathe, M., and A. K. S. Vullikanti. 2013. "Computational Epidemiology". *Communications of the ACM* 56(7):88–96.

Moosa, J., W. Awad, and T. Kalganova. 2023. "COVID-19 Contact-Tracing Networks Datasets". In *2023 International Conference on IT Innovation and Knowledge Discovery (ITIKD)*, 1–4. IEEE.

Newman, M. E. 2003. "The Structure and Function of Complex Networks". *SIAM review* 45(2):167–256.

Prakash, B. A., D. Chakrabarti, N. C. Valler, M. Faloutsos, and C. Faloutsos. 2012. "Threshold Conditions for Arbitrary Cascade Models on Arbitrary Networks". *Knowledge and Information Systems* 33:549–575.

PREPARE 2023. "Synthetic Pandemic Outbreaks". https://prepare-vo.org/synthetic-pandemic-outbreaks, accessed April 15th.

Qiu, Z., B. Espinoza, V. V. Vasconcelos, C. Chen, S. M. Constantino, S. A. Crabtree, L. Yang, A. Vullikanti, J. Chen, J. Weibull, K. Basu, A. Dixit, S. A. Levin, and M. V. Marathe. 2022. "Understanding the Coevolution of Mask Wearing and Epidemics: A Network Perspective". *Proceedings of the National Academy of Sciences* 119(26):e2123355119.

Richardson, C. 2018. *Microservices Patterns: With Examples in Java*. Simon and Schuster.

Rodríguez, P., S. Graña, E. E. Alvarez-León, M. Battaglini, F. J. Darias, M. A. Hernán, R. López, P. Llaneza, M. C. Martín, RadarCovidPilot Group et al. 2021. "A Population-Based Controlled Experiment Assessing the Epidemiological Impact of Digital Contact Tracing". *Nature Communications* 12(1):587.

Rozenshtein, P., A. Gionis, B. A. Prakash, and J. Vreeken. 2016. "Reconstructing an Epidemic Over Time". In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, 1835–1844. New York, NY, USA: Association for Computing Machinery.

Sahu, S., A. Mhedhbi, S. Salihoglu, J. Lin, and M. T. Özsu. 2020. "The Ubiquity of Large Graphs and Surprising Challenges of Graph Processing: Extended Survey". *The VLDB Journal* 29:595–618.

Shah, D., and T. Zaman. 2011, 09. "Rumors in a Network: Who's the Culprit?". *IEEE Transactions on Information Theory* 57:5163 – 5181.

Thorve, S., A. Vullikanti, S. Swarup, H. Mortveit, and M. Marathe. 2022. "Modular and Extensible Pipelines for Residential Energy Demand Modeling and Simulation". In *WSC'22: Proceedings of the Winter Simulation Conference*, edited by B. Feng, P. Yijie, G. Pedrielli, S. Eunhye, S. Shashaani, and C. G. Corlu, 855–866. IEEE.

Tsvyatkova, D., J. Buckley, S. Beecham, M. Chochlov, I. R. O'Keeffe, A. Razzaq, K. Rekanar, I. Richardson, T. Welsh, C. Storni, and COVIGILANT Group. 2022. "Digital Contact Tracing Apps for COVID-19: Development of a Citizen-Centered Evaluation Framework". *JMIR mHealth and uHealth* 10(3):e30691.

Verelst, F., L. Willem, and P. Beutels. 2016. "Behavioural Change Models for Infectious Disease Transmission: a Systematic Review (2010–2015)". *Journal of The Royal Society Interface* 13(125):20160820.

Wolff, E. 2016. *Microservices: Flexible Software Architecture*. Addison-Wesley Professional.

Wymant, C., L. Ferretti, D. Tsallis, M. Charalambides, L. Abeler-Dörner, D. Bonsall, R. Hinch, M. Kendall, L. Milsom, M. Ayres et al. 2021. "The Epidemiological Impact of the NHS COVID-19 App". *Nature* 594(7863):408–412.

AUTHOR BIOGRAPHIES

AMRO ALABSI ALJUNDI is a PhD student in the Department of Computer Science at the University of Virginia. His research interests are in high-performance computing and machine learning. His email address is nmm2uy@virginia.edu.

GALEN HARRISON is a PhD student in the Department of Computer Science at the University of Virginia. His research interests center the role of data in public decision making. His email address is gh7vp@virginia.edu.

JIANGZHUO CHEN is a Research Associate Professor in the Biocomplexity Institute and Initiative at the University of Virginia. His research interests are in computational epidemiology, modeling and simulation and causal machine learning. His email address is chenj@virginia.edu.

MADHAV MARATHE is a Distinguished Professor in Biocomplexity at the University of Virginia with interests in network science, computational epidemiology, AI, foundations of computing, socially coupled system science and high performance computing. His email address is marathe@virginia.edu.

HENNING S. MORTVEIT is an Associate Professor in the Biocomplexity Institute and Initiative and the Department of Engineering Systems and Environment at the University of Virginia with research interest in massively interacting systems, software design and computational architectures. His email address is Henning.Mortveit@virginia.edu.

ANIL VULLIKANTI is a Professor in the Biocomplexity Institute and Initiative and the Department of Computer Science. His interests are in network science, and the foundations of AI and machine learning. His email address is vsakumar@virginia.edu.

ABHIJIN ADIGA is a Research Associate Professor in the Biocomplexity Institute and Initiative at the University of Virginia. His research interests are in network science, foundations of AI, and modeling and simulation. His email address is abhijin@virginia.edu.